A. Additional Experimental Results

A.1. Variants of Weight with Hard Instance Selection

In Eq. (5) we propose a soft weight. Alternatively, we consider a hard weight, where top-k instances receive entire weight while weights for the rest of instances are set to 0. Specifically, let $I = \text{argsort}\left[\mathbb{E}_{j\neq i}\{\min_n \|\mathbf{z}_m^i - \mathbf{z}_n^j\|\}\right]$, a sorted list in descending order. Let I_k is a set of indices including the first k items of the list I. The hard weight is defined as:

$$\alpha_m^i = \frac{1}{k}\mathbf{1}\{m \in I_k\} \tag{8}$$

We present results in Figure 7b. Overall we observe a similar trend with the soft weight in Figure 7a. For example, both τ and k work robustly when their values are small for texture categories. For object categories we generally require a bit larger τ or k to obtain an optimal performance, as defective regions could be sometimes larger and even global. Between hard and soft weights, we find that soft weights are slightly better as it still assigns different weights to instances while hard weight assigns uniform weights to top-k instances. One could develop to take the best of both worlds as follows:

$$\alpha_m^i \propto \exp\left(\frac{1}{\tau}\mathbb{E}_{j\neq i}\{\min_n \|\mathbf{z}_m^i - \mathbf{z}_n^j\|\}\right),\ m \in I_k \ \text{ or 0 otherwise.} \tag{9}$$

A.2. Analysis on Labeled Normal Data Size

In this section we study the impact of the number of labeled normal data on the clustering performance of semi-supervised weighted average distance. Specifically, we vary the number of labeled normal data used to compute the weight of Eq. (6).

The summary results are in Figure 8. We also plot the performance of unsupervised version of Eq. (5). For object and MTD we find a clear trend of performance improvement as we increase the number of labeled normal data, while for texture the performance does not change much. Since acquiring labeled normal data is a lot cheaper than acquiring labeled anomaly data of multiple types, our results suggest a relatively inexpensive way to improve the clustering performance with a minimal supervision. For example, 20% of labeled normal data for object categories of MVTec dataset corresponds to around 50 images.

A.3. Patch vs Holistic Representation

We provide results comparing the clustering performance of holistic and patch-based representations using ResNet [24, 69] and EfficientNet [58] models. Specifically, we conduct experiments using ResNet with various depths (18, 50, 101, 152) and EfficientNet with various sizes (B0, B4, B7). Summary results are in Figure 9. For all accumulated bar plots over three datasets, we observe consistent trend of improved anomaly clustering performance using patch-based representations (second, third and fourth columns, with maximum Hausdorff, weighted average and semi-supervised version of that, respectively) over a holistic representation (first column).

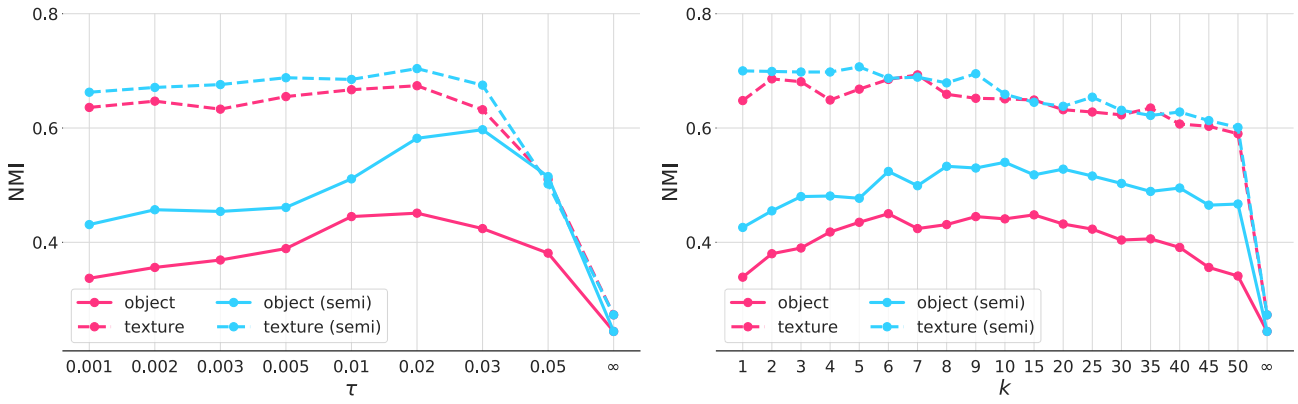
A.4. Feature Extractor

We study the performance of anomaly clustering for various feature extractors, including ResNet [24, 69], EfficientNet [58], and Vision Transformer (ViT) [16]. All aforementioned models are trained on ImageNet [14]. We provide which layer and average pooling kernel size have been used for each network in Table 6.

Table 6: Implementation details on the layer and average pooling kernel size used for each network architecture.

Network	ResNet	EfficientNet	ViT-T	ViT-S	ViT-B	ViT-L
Layer used	ResBlock 2	Reduction 3	Block 7			Block 13
Kernel size	3×3		1×1			

The results are in Figure 10 and 11. We plot accumulated NMI scores of average distance (i.e., $\alpha = \frac{1}{M}$), maximum Hausdorff, and weighted distance without and with labeled normal data. It is clear that the proposed multiple instance clustering framework outperforms a single instance clustering via average distance. We observe weighted average improves upon maximum Hausdorff for many cases. Moreover, semi-supervised version of weighted average distance significantly improves the performance.



(a) NMI-vs-τ (b) NMI-vs-k

Figure 7: Sensitivity analysis of τ and k on MVTec dataset.

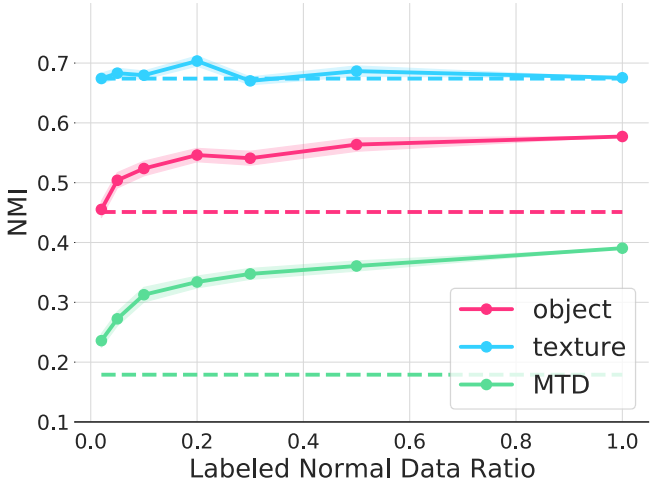


Figure 8: NMI scores of semi-supervised anomaly clustering with varying ratios of labeled normal data. Plots with dotted line represent unsupervised anomaly clustering results with the proposed weighted distance.

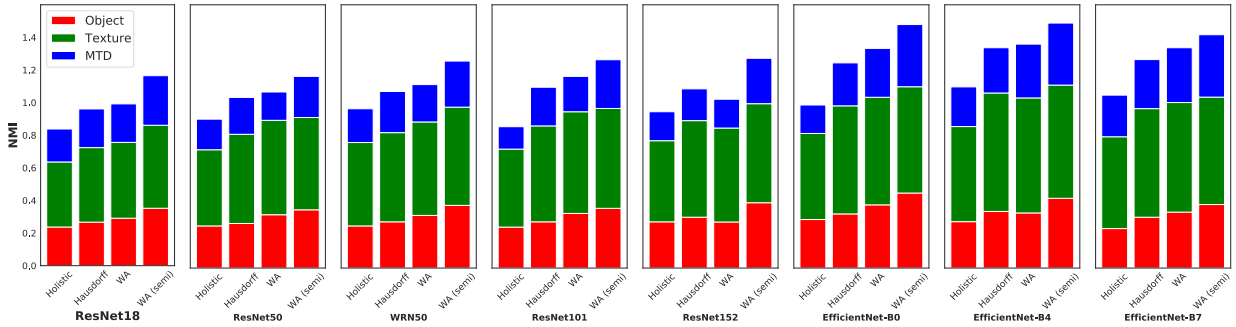


Figure 9: Bar plots with NMI scores over three datasets using various ResNet and EfficientNet models with last hidden layer. We show results for holistic and patch-based with maximum Hausdorff distance, weighted average distance, and its semi-supervised version.

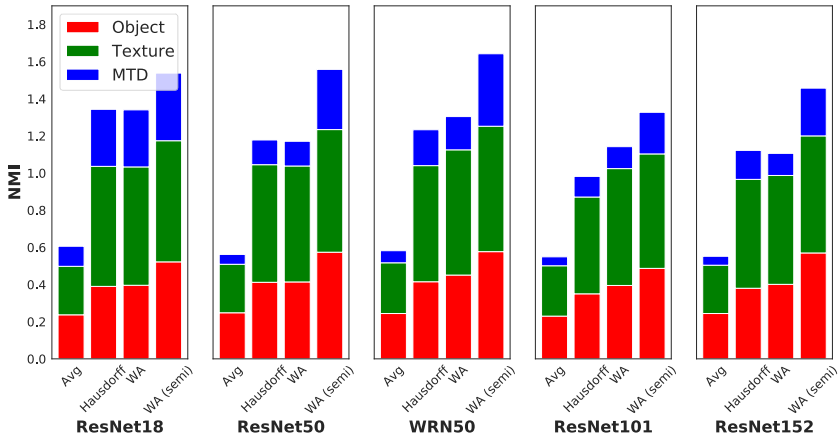


Figure 10: Bar plots with NMI scores over three datasets using various ResNet models with their intermediate layers as in Table 6. We show results for average, maximum Hausdorff distance, weighted average distance, and its semi-supervised version.

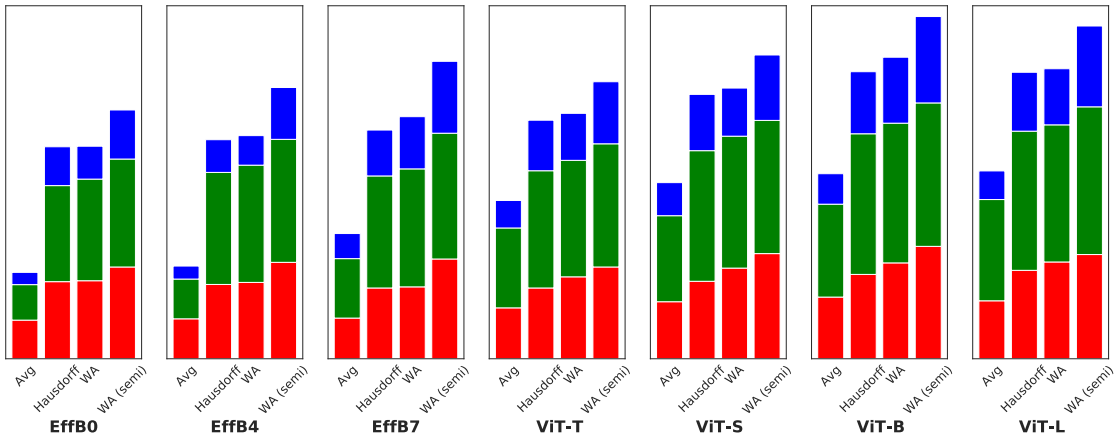


Figure 11: Bar plots with NMI scores over three datasets using various EfficientNet and ViT models with their intermediate layers as in Table 6. We show results for average, maximum Hausdorff distance, weighted average distance, and its semi-supervised version.

A.5. Results on Purity with Overclustering

We provide additional results on the purity of clusters with overclustering in Figure 13 on MVTec dataset. We also present the area under the curve divided by the total number of examples (mAUC) in the bracket of each legend. As we see in Figure 13, we observe significantly higher purity with our proposed clustering framework (brown, green, light blue) over the baseline (pink) for most cases.

B. Additional Analysis with Variants of Distance Measure

We provide additional qualitative reasons on why $\max$ or $\min$ operators perform less robust than $\mathbb{E}$ when computing unsupervised weights of Eq. (5). Firstly, the downside of $\min$ operator is clear from the formulation. To be clear, we write the formulation as follows:

$$\alpha_m^i \propto \exp\left(\frac{1}{\tau}\min_{j\neq i}\left\{\min_n \|\mathbf{z}_m^i - \mathbf{z}_n^j\|\right\}\right) \tag{10}$$

Let an image $\mathbf{x}_i$ is a duplicate of $\mathbf{x}_j$, i.e., $\mathbf{x}_i = \mathbf{x}_j$. Then, for any $\mathbf{z}_m^i$, we can always find $\mathbf{z}_n^j$ whose distance is 0. In other words, $\min_n \|\mathbf{z}_m^i - \mathbf{z}_n^j\| = 0$ for all m, and we get an uniform weight $\alpha_m^i \propto \exp(0)$. This is problematic if $\mathbf{x}_i$ is indeed an

anomalous image as α does not provide any meaningful signal to attend to the defective area.

Secondly, as in Figure 12, the $\max$ operator would highlight the blue cable as it does not exists for some images in the dataset, even though it is a normal pattern.

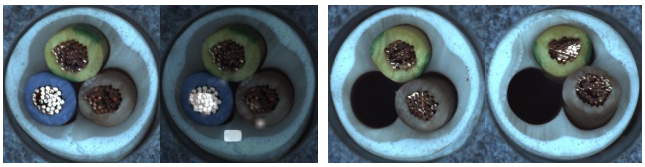


Figure 12: An input image $\mathbf{x}^i$ and that with weight overlaid when computed via $\max$ operators against $\mathbf{x}^j$'s on the right.

C. Formulations for Variant of Hausdorff Distance

In this section we provide exact formulations that we use for experiments in Section 5.3.

1. Eq. (3): mean mean, Eq. (2): – :

$$d_{\text{avgavg}}(Z_i, Z_j) = \frac{1}{MN} \sum_{m=1,\dots,M} \sum_{n=1,\dots,M} \{\|\mathbf{z}_m^i - \mathbf{z}_n^j\|\}$$

2. Eq. (3): max min, Eq. (2): max :

$$d_{\text{maxH}}(Z_i, Z_j) = \max\{d(Z_i, Z_j), d(Z_j, Z_i)\},$$
$$d(Z_i, Z_j) = \max_{m=1,\dots,M} \min_{n=1,\dots,M} \{\|\mathbf{z}_m^i - \mathbf{z}_n^j\|\}$$

3. Eq. (3): max min, Eq. (2): mean :

$$d_{\text{maxH}-\text{avg}}(Z_i, Z_j) = \frac{1}{2}\big(d(Z_i, Z_j) + d(Z_j, Z_i)\big),$$
$$d(Z_i, Z_j) = \max_{m=1,\dots,M} \min_{n=1,\dots,M} \{\|\mathbf{z}_m^i - \mathbf{z}_n^j\|\}$$

4. Eq. (3): min min, Eq. (2): – :

$$d_{\text{minmin}}(Z_i, Z_j) = \min_{m=1,\dots,M} \min_{n=1,\dots,M} \{\|\mathbf{z}_m^i - \mathbf{z}_n^j\|\}$$

5. Eq. (3): mean min, Eq. (2): max :

$$d_{\text{avgmin}}(Z_i, Z_j) = \max\{d(Z_i, Z_j), d(Z_j, Z_i)\},$$
$$d(Z_i, Z_j) = \frac{1}{M} \sum_{m=1,\dots,M} \min_{n=1,\dots,M} \{\|\mathbf{z}_m^i - \mathbf{z}_n^j\|\}$$

6. Eq. (3): mean min, Eq. (2): mean :

$$d_{\text{avgmin}}(Z_i, Z_j) = \frac{1}{2}\big(d(Z_i, Z_j) + d(Z_j, Z_i)\big),$$
$$d(Z_i, Z_j) = \frac{1}{M} \sum_{m=1,\dots,M} \min_{n=1,\dots,M} \{\|\mathbf{z}_m^i - \mathbf{z}_n^j\|\}$$

Table 7: Normalized mutual information (NMI), adjusted rand index (ARI) and F1 scores of unsupervised and semi-supervised clustering methods on MVTec (object, texture) and MTD datasets. Hierarchical Ward clustering is used for clustering, while various distance measures, such as average, maximum Hausdorff, or the proposed weighted average, are used to compute pairwise distances between data. We also report the performance of weighted average distance whose weights are generated from the ground-truth segmentation masks.

Supervision	Unsupervised									Semi (labeled normal data)			Segmentation mask		
Distance	Average			Maximum Hausdorff			Weighted Average			Weighted Average			Weighted Average		
Dataset	NMI	ARI	F1	NMI	ARI	F1	NMI	ARI	F1	NMI	ARI	F1	NMI	ARI	F1
bottle	0.426	0.186	0.448	0.585	0.510	0.764	0.495	0.421	0.567	**0.607**	0.461	0.639	0.531	0.438	0.584
cable	0.439	0.209	0.421	0.636	0.348	0.579	0.730	0.673	0.770	**0.889**	0.903	0.935	**0.939**	0.934	0.849
capsule	0.172	0.060	0.339	0.156	0.045	0.276	0.185	0.070	0.380	**0.334**	0.191	0.466	**0.443**	0.329	0.533
hazelnut	0.063	-0.003	0.314	0.552	0.327	0.500	0.568	0.430	0.610	**0.868**	0.889	0.936	**0.904**	0.925	0.954
metal nut	0.342	0.160	0.376	0.448	0.339	0.542	0.610	0.439	0.527	**0.639**	0.457	0.624	0.556	0.373	0.528
pill	0.313	0.134	0.300	0.384	0.169	0.390	0.469	0.246	0.419	**0.515**	0.317	0.438	**0.653**	0.484	0.576
screw	0.049	-0.000	0.264	0.031	-0.006	0.239	0.038	-0.007	0.251	**0.376**	0.267	0.418	**0.592**	0.505	0.672
toothbrush	0.000	-0.018	0.581	**0.251**	0.050	0.652	0.214	-0.008	0.599	0.214	-0.008	0.599	**1.000**	1.000	1.000
transistor	0.282	0.110	0.497	0.499	0.478	0.703	0.573	0.674	0.755	**0.651**	0.462	0.594	**0.825**	0.921	0.874
zipper	0.353	0.255	0.454	0.606	0.491	0.615	0.628	0.521	0.648	**0.677**	0.552	0.635	**0.800**	0.614	0.679
carpet	0.287	0.138	0.392	**0.660**	0.586	0.795	0.656	0.576	0.647	0.550	0.430	0.553	**0.707**	0.592	0.614
grid	0.158	0.033	0.326	0.129	0.018	0.308	0.143	0.018	0.304	**0.258**	0.093	0.361	0.137	0.019	0.312
leather	0.398	0.218	0.465	0.725	0.652	0.762	0.778	0.674	0.704	**0.787**	0.677	0.728	0.712	0.632	0.684
tile	0.288	0.157	0.444	0.932	0.914	0.957	**0.933**	0.921	0.957	0.930	0.922	0.957	**1.000**	1.000	1.000
wood	0.231	0.066	0.384	0.678	0.500	0.716	**0.860**	0.815	0.921	0.823	0.725	0.893	**0.868**	0.802	0.907
MTD	0.065	0.024	0.289	0.193	0.112	0.381	0.179	0.120	0.346	**0.390**	0.314	0.490	**0.467**	0.359	0.482

C.1. Implementation Details for Deep Clustering

We follow general guidelines provided by the authors of IIC [28],[3] GATCluster [44],[4] and SCAN [60],[5] for experiments with deep clustering methods. For IIC and SCAN, we use a ResNet-50 backbone. We replace the first step of the SCAN, which is the self-supervised pretraining, with an ImageNet pretraining as the number of images for each dataset we consider in the paper is relatively small (e.g., 100$\sim$1000, as opposed to 50k for CIFAR-10 or more than a million for ImageNet). For GATCluster, we use the custom CNN architecture suggested by the author for ImageNet experiments.

For hyperparameters, we simply use the ones suggested by the authors. While these hyperparameters may not be optimal for anomaly detection datasets, we believe this is fair treatment as we do not conduct serious hyperparameter tuning for our methods.

[3] https://github.com/xu-ji/IIC
[4] https://github.com/niuchuangnn/GATCluster
[5] https://github.com/wvangansbeke/Unsupervised-Classification

Table 8: Comparison to other clustering methods, including classic clustering methods such as KMeans, GMM, spectral, or hierarchical clustering with various linkages, using maximum Hausdorff (maxH) or weighted average (WA) distances, and deep clustering methods, such as IIC [28], GATCluster [44], or SCAN [60]. For deep clustering methods, we also provide in the parenthesis the performance of the best training epoch chosen by the test set accuracy.

Dataset	MVTec Object						MVTec Texture						Magnetic Tile Defect					
Distance	maxH			WA			maxH			WA			maxH			WA		
Metric	NMI	ARI	F1	NMI	ARI	F1	NMI	ARI	F1	NMI	ARI	F1	NMI	ARI	F1	NMI	ARI	F1
KMeans	–			0.429	0.301	0.544	–			0.642	0.567	0.714	–			**0.204**	0.135	0.374
GMM	–			0.395	0.264	0.498	–			0.578	0.469	0.635	–			**0.204**	0.141	0.377
Spectral	0.419	0.287	0.546	0.428	0.305	0.555	0.609	0.525	0.702	0.606	0.516	0.681	0.143	0.089	0.354	0.150	0.098	0.341
Single	0.108	0.025	0.238	0.133	0.041	0.261	0.078	0.008	0.173	0.108	0.005	0.186	0.087	0.019	0.202	0.065	0.012	0.200
Complete	0.316	0.187	0.409	0.294	0.146	0.405	0.360	0.184	0.356	0.452	0.265	0.510	0.128	0.062	0.320	0.116	0.075	0.310
Average	0.245	0.109	0.328	0.276	0.095	0.345	0.223	0.064	0.294	0.400	0.213	0.398	0.080	0.024	0.242	0.094	0.034	0.284
Ward	0.415	0.275	0.526	**0.451**	0.346	0.553	0.625	0.534	0.708	**0.674**	0.601	0.707	0.193	0.112	0.381	0.179	0.120	0.346

	NMI	ARI	F1	NMI	ARI	F1	NMI	ARI	F1
IIC	0.086 (0.170)	0.019 (0.117)	0.297 (0.366)	0.107 (0.188)	0.023 (0.096)	0.261 (0.300)	0.064 (0.034)	0.020 (0.017)	0.252 (0.230)
GATCluster	0.119 (0.265)	0.044 (0.202)	0.320 (0.475)	0.171 (0.298)	0.072 (0.202)	0.305 (0.442)	0.028 (0.113)	0.009 (0.064)	0.243 (0.333)
SCAN	0.176 (0.198)	0.078 (0.123)	0.335 (0.393)	0.277 (0.314)	0.153 (0.203)	0.335 (0.393)	0.071 (0.087)	0.029 (0.053)	0.282 (0.309)

Table 9: NMI, ARI and F1 scores of unsupervised and semi-supervised clustering methods on MVTec (object, texture) datasets. Compared to the baseline method ("average") that uses a holistic representation via average pooling of patch embeddings, the multiple instance clustering framework with various distance measures, such as maximum Hausdorff or the proposed weighted average distances, show huge improvement. We also report the performance of weighted average distance whose weights are computed using labeled normal data ("Semi"). Furthermore, we include extended baselines using max pooling, generalized mean pooling (GeM), sum-pooled convolutional features (SPoC) [1], and bag-of-words with the codebook size of 512. We test each method on the random subsets including 90% images of the test set for 100 different random seeds to compute mean and standard errors.

Supervision	Unsupervised									Semi		
Metric	NMI	ARI	F1	NMI	ARI	F1	NMI	ARI	F1	NMI	ARI	F1
Distance	Average			Maximum Hausdorff			Weighted Average			Weighted Average		
MVTec (object)	0.249	0.114	0.412	0.423	0.274	0.520	0.458	0.333	0.563	0.584	0.477	0.653
std err.	(0.002)	(0.003)	(0.004)	(0.004)	(0.005)	(0.004)	(0.003)	(0.005)	(0.004)	(0.004)	(0.006)	(0.005)
MVTec (texture)	0.288	0.122	0.405	0.650	0.560	0.722	0.665	0.582	0.709	0.702	0.616	0.743
std err.	(0.003)	(0.003)	(0.003)	(0.004)	(0.005)	(0.005)	(0.003)	(0.004)	(0.003)	(0.004)	(0.005)	(0.004)
Distance	Max			GeM ($p = 20$)			SPoC ($\sigma = 1000$)			Bag-of-Words		
MVTec (object)	0.336	0.204	0.488	0.338	0.209	0.486	0.249	0.114	0.412	0.226	0.102	0.396
std err.	(0.003)	(0.004)	(0.004)	(0.003)	(0.004)	(0.004)	(0.002)	(0.003)	(0.004)	(0.003)	(0.003)	(0.003)
MVTec (texture)	0.598	0.478	0.658	0.602	0.482	0.660	0.288	0.122	0.405	0.312	0.126	0.359
std err.	(0.004)	(0.004)	(0.003)	(0.003)	(0.004)	(0.003)	(0.003)	(0.003)	(0.003)	(0.004)	(0.004)	(0.004)

Table 10: Comparison to other clustering methods, including KMeans, KMedoids, GMM, spectral, and hierarchical clustering with various linkages, using maximum Hausdorff (maxH) or weighted average (WA) distances, and deep clustering methods, such as IIC [28], GATCluster [44], or SCAN [60]. For deep clustering methods, we provide in the parenthesis the performance of the best training epoch chosen by test set accuracy. We test each method on the random subsets including 90% images of the test set for 100 different random seeds to compute mean and standard errors.

Dataset	MVTec (object)		MVTec (texture)		MTD	
Distance	maxH	WA	maxH	WA	maxH	WA
KMeans	–	0.429±0.002	–	0.637±0.002	–	**0.204**
GMM	–	0.397±0.002	–	0.583±0.003	–	**0.204**
KMedoids	0.152±0.005	0.250±0.004	0.301±0.005	0.391±0.006	0.050	0.076
Spectral	0.415±0.003	0.422±0.002	0.618±0.003	0.606±0.003	0.143	0.150
Single	0.122±0.003	0.141±0.003	0.086±0.002	0.116±0.002	0.087	0.065
Complete	0.321±0.005	0.339±0.005	0.404±0.007	0.495±0.007	0.128	0.116
Average	0.225±0.005	0.213±0.002	0.272±0.007	0.367±0.007	0.080	0.094
Ward	0.423±0.004	**0.458**±0.003	0.650±0.004	**0.665**±0.003	0.193	0.179
IIC	0.086 (0.170)		0.107 (0.188)		0.064 (0.034)	
GATCluster	0.119 (0.265)		0.171 (0.298)		0.028 (0.113)	
SCAN	0.176 (0.198)		0.277 (0.314)		0.071 (0.087)	

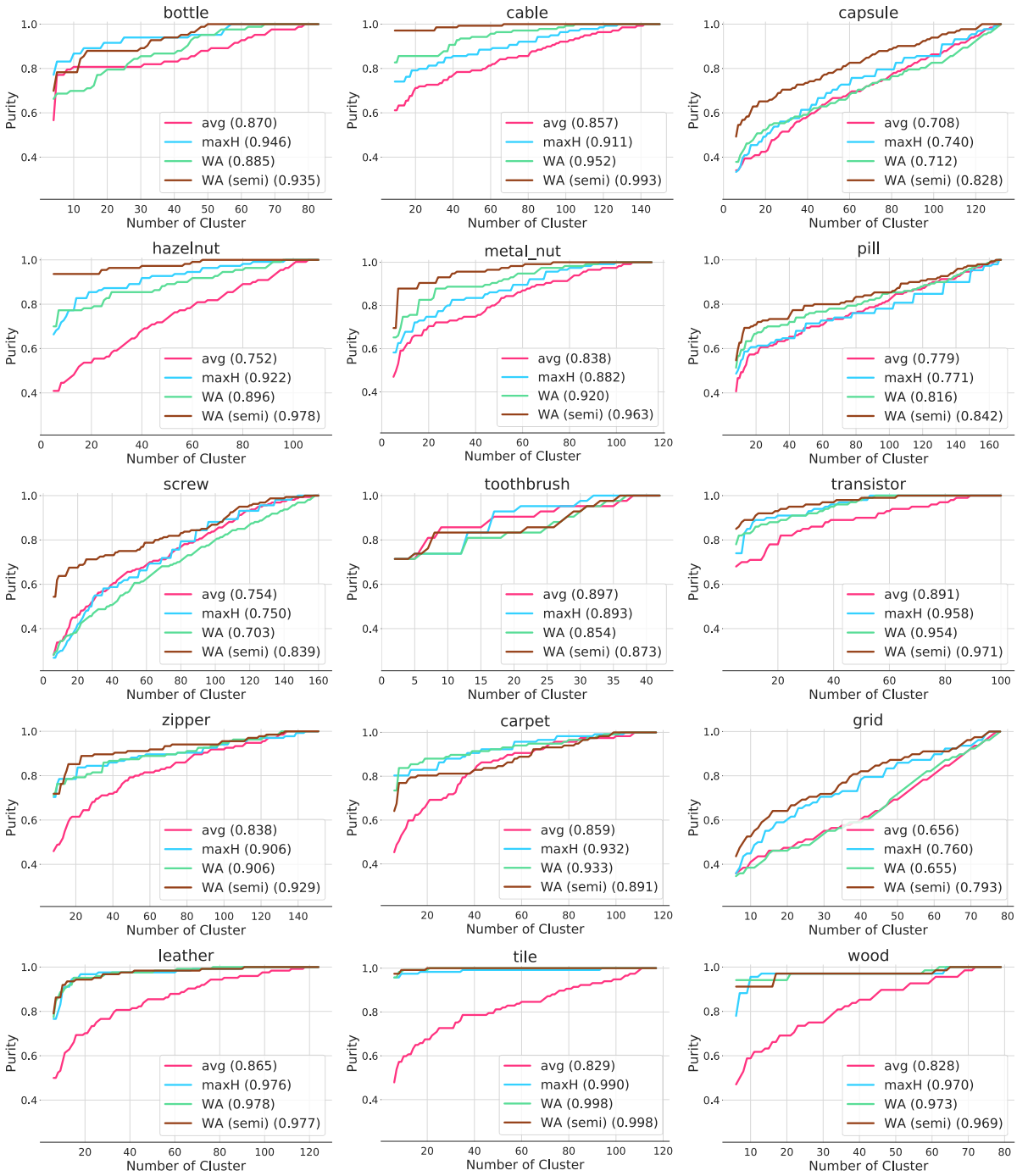


Figure 13: Purity of clusters with different number of clusters on MVTec dataset. Hierarchical Ward clustering is used for clustering method with different attention strategies including uniform, top-k, and soft. Numbers in the bracket represent the area under the curve divided by the total number of examples.